

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

UTILITY PATENT APPLICATION FOR:

**METHOD AND APPARATUS FOR STORING REAL-TIME  
TEXT MESSAGES**

Inventor:

Marina LIBMAN  
7915 Crisford Place, Apartment L  
Pikesville, MD 21208

Docket No.: 003636.0067

# PATENT

## METHOD AND APPARATUS FOR STORING REAL-TIME TEXT MESSAGES

### FIELD OF THE INVENTION

This invention relates generally to a system for transferring data, e.g., real-time text messages, between a source and destination computing platforms. More specifically, a synchronization module is utilized to both transfer and convert data, e.g., archived real-time text messages, from a format used by the source computing platform into a format suitable for use by the destination computing platform.

### DESCRIPTION OF THE RELATED ART

Real time text messaging ("instant messaging") has gained widespread popularity. Instant messaging may be a valuable business tool because it allows users to communicate messages almost instantaneously to each other. Traditional computing platforms (e.g., personal computers, laptops, and the like) typically include a feature which enables a user to save instant messages (e.g., conversations, chat histories, etc.) in permanent memory of a computing system. Additionally, a user may be able to select a directory on a file system in permanent memory and/or a file format (e.g., ASCII, rich text, HTML, etc.) for saving the instant messages.

With the technological advances of wireless devices (e.g., hand held computers, web capable wireless telephones, etc.), the ability to perform instant messaging has been extended beyond the conventional computing platforms. In this respect, it is now possible for users to perform instant messaging from virtually anywhere by employing any number of wireless devices.

While instant messaging with wireless devices is convenient, wireless devices are not quite as convenient as the conventional computing platforms. For instance, the capability of conventional computing platforms to archive instant messages may be limited due to the relatively small amounts of memory typically found in conventional wireless devices. Accordingly, the relatively small amount of memory available on wireless devices may somewhat limit the number and length of instant messages that may be archived in the memory of the wireless devices. Additionally, because the wireless devices typically archive the instant messages under relatively unique formats, inter-device compatibility and availability may be limited.

## SUMMARY OF THE INVENTION

According to one aspect, the present invention relates to a method for transferring data between a data source and a data sink. The method includes initiating a transfer of a message history data and transferring the message history data in response to an establishment of a communication channel. The method further includes converting received message history data to a previously selected data format and storing converted message history data in a previously selected location.

In another aspect, the present invention relates to a method for transferring chat history. The method includes initiating a transfer of the chat history and transferring the chat data in response to an establishment of a communication channel. The method further includes combining associated data related to the chat history and determining a destination of the chat history.

1 In yet another aspect, the present invention relates to a method for synchronizing a  
2 message history. The method includes initiating a transfer of the message history and  
3 transferring the message history in response to an establishment of a communication channel.  
4 The method includes determining a destination of the message history.

5 In yet another aspect, the present invention relates to an apparatus for synchronizing a  
6 chat history. The apparatus includes an interface adapted to communicate with a destination  
7 device and a memory configured to store the chat history of a messaging program. The  
8 apparatus also includes a processor configured to accept a synchronization request, and to  
9 transfer the chat history in response to an establishment of a communication channel through  
10 the interface.

11 In yet another aspect, the invention relates to a source device for synchronizing a  
12 message history. The source device includes the message history of a messaging program.  
13 The source device also includes a processor configured to accept a synchronization request,  
14 and to transfer the message history in response to an establishment of a communication  
15 channel through the interface.

16 In yet another aspect, the invention relates to a destination device for synchronizing a  
17 message history. The destination device includes an interface adapted to communicate with a  
18 source device and a memory. The destination device also includes a processor configured to  
19 establish a communication channel with the source device through the interface in response to  
20 a synchronization request at the source device and to activate a synchronization module  
21 configured to accept the message history from the source device in response to an activation  
22 message from the source device.

1 In yet another aspect, the invention relates to a system for synchronizing a chat  
2 history. The system includes a communication network, a source device configured to  
3 transfer the chat history over the communication network and a destination device configured  
4 to receive the chat history. The system also includes a source synchronization module  
5 executing on the source device and a destination synchronization module adapted to execute  
6 on the destination device, where the source synchronization module is configured to transfer  
7 the chat history in response to an activation of the destination synchronization module by the  
8 source synchronization module.

9 In yet another aspect, the invention relates to a computer readable storage medium on  
10 which is embedded one or more computer programs. The one or more computer programs  
11 implement a method of transferring a message history data, where the one or more computer  
12 programs include a set of instructions for initiating a transfer of the chat history and  
13 transferring the chat data in response to an establishment of a communication channel. The  
14 one or more computer programs also include a set of instructions for combining associated  
15 data related to the chat data and determining a destination of the chat data.

16 In yet another aspect, the invention relates to a computer readable storage medium on  
17 which is embedded one or more computer programs. The one or more computer programs  
18 implement a method of transferring a chat history, where the one or more computer programs  
19 include a set of instructions for transferring the chat history in response to an establishment of  
20 a communication channel with the destination and converting received data to a previously  
21 selected data format, and storing converted chat history in a previously selected location.

22 In yet another aspect, the invention relates to a computer readable storage medium on

1 which is embedded one or more computer programs. The one or more computer programs  
2 implement a method of synchronizing a message history, where the one or more computer  
3 programs include a set of instructions for initiating a transfer of the message history,  
4 transferring the message history in response to an establishment of a communication channel,  
5 and determining a destination of the message history.

## 6 7 BRIEF DESCRIPTION OF THE DRAWINGS

8 Features and advantages of the present invention will become apparent to those skilled  
9 in the art from the following description with reference to the drawings, in which:

10 Fig. 1 illustrates an exemplary block diagram of a system architecture according to the  
11 principles of the present invention;

12 Fig. 2 illustrates an exemplary block diagram of a wireless device as a source  
13 computing platform shown in Fig. 1 in accordance with the principles of the present  
14 invention;

15 Fig 3. illustrates an exemplary block diagram of a software architecture for the  
16 wireless device shown in Fig. 2 in accordance with the principles of the present invention;

17 Fig. 4 illustrates an exemplary block diagram of a computer system as the destination  
18 computing platform, as shown in Fig. 1, in accordance with the principles of the present  
19 invention;

20 Fig. 5 illustrates an exemplary block diagram of a software architecture for the  
21 destination computer system, as shown in Fig. 4, in accordance with the principles of the  
22 present invention;

Fig. 6 illustrates an exemplary flow diagram of a synch module functioning as a source synch module for either of the synchronization modules as shown in Figs. 3 and 5, respectively, in accordance with the principles of the present invention;

Fig. 7 illustrates an exemplary flow diagram of the destination synchronization module portion of the synchronization modules of Figs. 3 and 5, respectively, in accordance with the principles of the present invention;

Fig. 8 illustrates an exemplary block diagram of a system architecture of another embodiment of the present invention;

Fig. 9 illustrates an exemplary block diagram of a software architecture of an embodiment of the present invention as implemented in the system architecture of Fig. 8;

Fig. 10 illustrates an exemplary flow diagram for a source synchronization module in the system architecture as shown in Figs. 8-9; and

Figs. 11A-C, together, illustrates an exemplary flow diagram for a destination/intermediate synchronization module in the system architecture as shown in Figs. 8-9.

## DETAILED DESCRIPTION OF THE INVENTION

For simplicity and illustrative purposes, the principles of the present invention are described by referring mainly to exemplary embodiments thereof, particularly with references to a synchronization module ("synch module"). However, one of ordinary skill in the art would readily recognize that the same principles are equally applicable to, and can be

1 implemented in, any type of computing environment capable of transferring data, and that any  
2 such variation would be within such modifications that do not depart from the true spirit and  
3 scope of the present invention. Moreover, in the following detailed description, references  
4 are made to the accompanying drawings, which illustrate specific embodiments in which the  
5 present invention may be practiced. Electrical, mechanical, logical and structural changes  
6 may be made to the embodiments without departing from the spirit and scope of the present  
7 invention. The following detailed description is, therefore, not to be taken in a limiting sense  
8 and the scope of the present invention is defined by the appended claims and their  
9 equivalents.

10 In accordance with the principles of the present invention, a synch module may reside  
11 on a source computing platform and a destination computing platform. The source computing  
12 platform may be one of the group comprising a wireless device (e.g., a text-messaging  
13 enabled pager, a personal digital assistant, a wireless telephone, etc.), a laptop computer, a  
14 desktop computer, or other similar computing platform. The destination computing platform  
15 may be selected from the same group of devices listed hereinabove for the source computing  
16 platform. However, the destination computing platform is not required to be identical to the  
17 source computing platform.

18 The synch module may be configured to operate as a source synch module or a  
19 destination synch module depending on which computing platform functions as a data source.

20 As a data source or initiator, the synch module may be configured to transfer data, e.g., an  
21 archived message history, from a source computing platform to a destination computing  
22 platform. At the destination computing platform, the synch module may be further



1 configured to convert the received archived message history to a specified data format, where  
2 the data format is predetermined by a user. The converted data, e.g., an archived message  
3 history, may then be stored in a pre-determined location by the synch module.

4 In particular, the synch module, functioning as an initiator (or source) of data on a  
5 source computing platform, may be configured to query for the availability of a destination  
6 computing platform. The synch module may be further configured to transfer data, e.g., an  
7 archived message history, in response to an establishment of a communication channel  
8 between the source computing platform and the destination computing platform. The  
9 communication may be implemented using conventional means such as wired (e.g., cable,  
10 modem, network, etc.) or wireless (e.g., infrared, optical, radio-frequency, etc.,) forms of  
11 communication. The synch module may be further configured to store the transferred data,  
12 e.g., an archived message history, to a destination synchronization module of the destination  
13 computing platform.

14 The synch module on the destination computing platform, functioning as a destination  
15 (or data sink), may be configured to convert the received data into a format specified by a  
16 user. The synch module may be configured to provide the capability to select a preferred or  
17 user-specified data format (e.g., text, HTML, PDF, etc.,) via a user-interface such as a menu  
18 option, a command line prompt, an icon, etc. The synch module may be further configured  
19 to store the converted data in a location also specified by the user via a user interface such as  
20 a dialog box, a command line prompt, an icon, etc.

21 In another aspect of the present invention, a synch module may reside on a source  
22 computing platform, an intermediate computing platform and a destination computing

1 platform. Although the synch module is substantially similar on each computing platform,  
2 the synch module may function as a source synch module, an intermediate synch module or a  
3 destination synch module depending on the role of the respective computing platform. In  
4 particular, the synch module residing on the source computing platform may function as a  
5 source synch module. The synch module residing on the intermediate computing platform  
6 may function as an intermediate synch module. The synch module residing on the destination  
7 computing platform may function as the destination synch module.

8 In the source synch module aspect, the synch module may be configured to receive a  
9 synchronization request from a user. The synchronization request, broadly speaking, may be  
10 configured to initiate a data transfer and conversion from one computing platform to another  
11 computing platform. The synch module may be further configured to establish a  
12 communication channel with an intermediate or destination computing platform. Once the  
13 communication channel is established, the synch module may be further configured to  
14 activate the synch module of the intermediate/destination computing platform. The synch  
15 module may be further configured to transfer data, e.g., an archived message history, to the  
16 intermediate/destination computing platform. The synch module of the  
17 intermediate/destination computing platform may be configured to combine any data related  
18 to the transferred data into a single file.

19 In the intermediate synch module aspect, the synch module of the  
20 intermediate/destination computing platform may be further configured to determine if the  
21 intermediate computing platform is the final destination of the data from the source  
22 computing platform. If the intermediate computing platform is the final destination, the synch

1 module may be configured to convert the transferred data into a data format specified by the  
2 user. The synch module of the intermediate computing platform may be further configured to  
3 send a completion signal or message to inform to the synch module of the source computing  
4 platform that the synchronization has been completed. Otherwise, the synch module of the  
5 intermediate computing platform may be further configured to determine whether the  
6 destination computing platform is available.

7 If the destination computing platform is unavailable, the synch module of the  
8 intermediate computing platform may temporarily store the transferred data while periodically  
9 attempting to establish a communication channel with the destination computing platform.  
10 Otherwise, the synch module may be further configured to establish a communication channel  
11 with the destination computing platform. The synch module may be further configured to  
12 activate the synch module of the destination computing platform in response to the  
13 establishment of a communication channel between the intermediate and destination  
14 computing platforms. The synch module of the intermediate computing platform may be  
15 further configured to transfer the designated data to the destination computing platform.

16 In the destination synchronization module aspect, the synch module of the destination  
17 computing platform may be configured to convert the received data into a data format  
18 selected by a user. The synch module may be further configured to store the converted data  
19 into a location also specified by the user.

20 Fig. 1 illustrates an exemplary block diagram of a system architecture 100 according  
21 to the principles of the present invention. As shown in Fig. 1, a source computing platform  
22 110 may be interfaced with a destination computer platform 120. The source computing

1 platform 110 may be a wireless device, e.g., a text-messaging pager, a wireless telephone, a  
2 personal digital assistant, etc., or a wired device such as a laptop computer, a computer, a  
3 server with clients, etc.

4 The destination computer platform 120 may be configured to provide a computing  
5 platform for applications and input/output services such as printing, information storage, etc.

6 The destination computer platform 120 may be a wireless device, e.g., a text-messaging  
7 pager, a wireless telephone, a personal digital assistant, etc., or a wired device such as a  
8 laptop computer, a computer, a server with clients, etc.

9 The destination computer platform 120 may be further configured to interface with a  
10 data storage device 125. The data storage device 125 may be configured to provide additional  
11 storage for information for the computer system 120. The data storage device 125 may be  
12 implemented by a disk drive, tape drive, or other type of permanent memory.

13 The source computing platform 110 may be configured to communicate with the  
14 destination computing platform 120 via wired and/or wireless methods. The wired methods  
15 may include a cable, a modem connection through the public switched telephone network,  
16 etc. The wireless method may include infrared, Bluetooth, radio frequency, etc. The source  
17 computing platform 110 and the destination computing platform 120 may communicate using  
18 a communication protocol such as Bluetooth, IEEE 802.3, REFLEX, etc.

19 The source and destination computing platforms, 110 and 120, may be configured to  
20 provide the capability of a messaging service such as instant messaging for their respective  
21 users. Accordingly, a user of the source computing platform 110 and/or destination  
22 computing platform 120 may use a supporting network of the source computing platform 110

1 to transmit and/or receive instant messages from other users. The other users may use a  
2 complementary device to the source computing platform 110 and/or destination computing  
3 platform 120 (e.g., PDAs, laptop computers, desktop computers, etc.,) to communicate. The  
4 messaging service (not shown) may be implemented as a software program executed by the  
5 source computing platform 110 and/or destination computing platform 120. The messaging  
6 service may be further configured to maintain a history of the messages, i.e., a log of  
7 messages, passed between the user and other users. The message history may be stored as  
8 text file, compressed data file, or in any format suitable for the source computing platform  
9 110 and/or destination computing platform 120.

10 The source computing platform 110 and destination computing platform 120 may be  
11 configured to execute a first and second synch module 130 and 130', respectively. First synch  
12 module 130 and second synch module 130' are substantially similar, and may be identical.  
13 The functionality exhibited by each synch module, 130 or 130', as a source synch module or a  
14 destination synch module is dependent on which computing platform is the source of the data  
15 transmission.

16 For example, if the source computing platform 110 is configured to the source of the  
17 data to be transferred and synchronized with the destination computing platform 120, the first  
18 synch module 130 of the source computing platform 110 may be configured to receive a  
19 synchronization request from a user. The first synch module 130 may be further configured  
20 to transfer data, e.g., a message history, to the second synch module 130' of the destination  
21 computing platform 120 over a communication channel. Although, for illustrative purposes  
22 only, Fig. 1 shows the source computing platform 110 as the source of data, it should be

1 readily apparent to those skilled in the art that either one of the source or destination  
2 computing platforms, 110 and 120, respectively, may be the source of data. Accordingly, a  
3 user initiates a synch command from the destination computing platform, the destination  
4 synch module attempts to establish the connection with the source synch module and “pulls”  
5 the message history from the source computing platform.

6 The second synch module 130' of the destination computing platform 120 may be  
7 configured to convert the received data into a predetermined data format by the user. The  
8 second synch module 130' may be further configured to store the converted message history  
9 in a predetermined location.

10 Conversely, if the second synch module 130' receives a synchronization request, the  
11 second synch module 130' may be configured to transfer data, e.g., a message history, to the  
12 first synch module 130 over a communication channel. The first synch module 130 may be  
13 configured to configured to convert the received data into a data format predetermined by the  
14 user. The first synch module 130 may be further configured to store the converted message  
15 history in a predetermined location.

16 Fig. 2 illustrates an exemplary block diagram of a wireless device 200 as the source  
17 computing platform 110 shown in Fig. 1 in accordance with the principles of the present  
18 invention. As shown in Fig. 2, the wireless device 200 may include a controller 205, a  
19 wireless interface 210, a memory 215, a display 220, a keypad 225, and a read-only memory  
20 230. Although, for illustrative purposes only, Fig. 2 illustrates the wireless device 200 with  
21 the above-mentioned components, it should be readily apparent to those of ordinary skill in  
22 the art that Fig. 2 represents a generalized schematic illustration of a wireless device and that

1 other components may be added or existing components may be subtracted without departing  
2 from the spirit or scope of the present invention.

3 The controller 205 may be configured to provide a computing platform for the  
4 software/firmware that provides the functionality of the wireless device 200 and to execute a  
5 computer program embodiment of the synch module 130. The controller 205 may be  
6 implemented with a microprocessor, a micro-controller, an application specific integrated  
7 circuit ("ASIC") a digital signal processor, and the like.

8 The controller 205 may be interfaced with a wireless interface 210. The wireless  
9 interface 210 may be configured to provide the capability for the user to receive and transmit  
10 information via radio frequency techniques such as cellular networks, pager networks, etc.  
11 The wireless interface 210 may be implemented by a pager interface, a wireless modem, a  
12 personal communication services ("PCS") interface, and the like.

13 The controller 205 may be further configured to interface with the memory 215. The  
14 memory 215 may be configured to provide the capability to store information and to provide  
15 application memory space. The memory 215 may be implemented with a random access  
16 memory ("RAM"), flash memory, or other re-writable memory.

17 The display 220 may be configured to interface with the controller 205. The display  
18 220 may be further configured to provide an interface for information to be displayed,  
19 including messages. The display 220 may include menu icons, program icons, or similar user  
20 interfaces to enable navigation through the functionality of the wireless device 200. The  
21 display 220 may be implemented with a liquid crystal display, thin film target display, or the  
22 like.

1 The controller 205 may be configured to interface with the keypad 225 which is  
2 configured to provide an interface for the performance of predetermined functions of the  
3 wireless device 200. The keypad 225 may be implemented by a numeric keypad, an  
4 alphanumeric keypad, or the like.

5 The read-only memory 230 of the wireless device 200 may be configured to provide  
6 non-volatile storage of software for the wireless device 200. The software may include an  
7 operating system, operational software that provides the functionality of the wireless device  
8 200, third party software, and the like. The read-only memory 230 may be implemented with  
9 an EEPROM, a PROM, a flash memory, or other types of permanent memories.

10 Fig 3. illustrates an exemplary block diagram of a software architecture 300 for the  
11 wireless device 200 shown in Fig. 2 in accordance with the principles of the present  
12 invention. As shown in Fig. 3, the software architecture 300 includes at least an operating  
13 system 310, an application program interface ("API") 315, a messaging program 320, and the  
14 synch module 130. Although, for illustrative purposes only, Fig. 3 illustrates the wireless  
15 device 200 with above-mentioned software architecture, it should be readily apparent to those  
16 of ordinary skill in the art that Fig. 3 represents a generalized illustration of the software  
17 architecture and that other software components may be added or existing software  
18 components may subtracted without departing from the spirit or scope of the present  
19 invention.

20 The operating system 310 of the wireless device 200 may be configured to manage  
21 software programs or applications that provide functionality to the wireless device 200. The  
22 operating system 310 may be further configured to provide an interface to the software



1 applications and to the underlying hardware components, as shown in Fig. 2, of the wireless  
2 device 200 through the API 315..

3 The messaging program 320 may be configured to provide the wireless device 200  
4 with the capability of receiving and transmitting instant messages between the user and other  
5 similarly-enabled users. The messaging program 320 may be configured to maintain a  
6 message history of conversations between the user and other similarly-enabled users. The  
7 message history may be stored on the memory 215 of the wireless device 200, shown in Fig.  
8 2. The messaging program 320 may be further configured to save the message history as a  
9 text file, a compressed data file, or in a format configured to conserve space in the memory  
10 215. Examples of a messaging program may include AETHER INSTANT MESSAGING,  
11 JABBER, etc.

12 The synch module 330 of the wireless device 200 may be configured to include the  
13 functionality of the first and second synch modules 130 or 130', respectively, as described  
14 herein above. Moreover, the synch module 330 may be configured to function an initiator of  
15 the transfer of data. Accordingly, the synch module 330 may be configured to transfer data,  
16 e.g., the saved or archived message history, from the memory 215 to a destination computer  
17 platform in response to a synchronization request. The synchronization request may be  
18 initiated, for example, as a menu option, an icon or manually initiated through the keypad  
19 225. The synch module 330 may be further configured to determine whether a  
20 communication channel is available over the wireless interface 210 to a destination computer  
21 system. In the event that the synch module 330 is unable to open a communication channel to  
22 a destination computer platform, the synch module 330 may be further configured to indicate

1 that the synchronization could not be performed. Alternatively, the synch module 330 of the  
2 source computing platform may be configured to receive data, e.g., a message history, from  
3 another computing platform, which is described in greater detail herein below.

4 Fig. 4 illustrates an exemplary block diagram of a computer system 400 as the  
5 destination computing platform 120 and executing a computer program embodiment of the  
6 synch module, 130 or 130', as shown in Fig. 1, in accordance with the principles of the  
7 present invention. Moreover, the computer system 400 may be configured to function a  
8 destination for the transfer of data from wireless device 200.

9 The functions of the synch modules, 130 or 130', may be implemented in program  
10 code and executed by the computer system 400. In particular, the computer system 400  
11 includes one or more processors, such as processor 402 that provides an execution platform  
12 for the destination synchronization module. Commands and data from the processor 402 are  
13 communicated over a communication bus 404. The computer system 400 also includes a  
14 main memory 406, preferably Random Access Memory (RAM), where the software for the  
15 destination synchronization module is executed during runtime, and a secondary memory 408.

16 The secondary memory 408 includes, for example, a hard disk drive 410 and/or a removable  
17 storage drive 412, representing a floppy diskette drive, a magnetic tape drive, a compact disk  
18 drive, etc., where a copy of software for the synch modules, 130 or 130', may be stored. The  
19 removable storage drive 412 reads from and/or writes to a removable storage unit 414 in a  
20 well-known manner. The synch modules, 130 or 130', is interfaced with a keyboard 416, a  
21 mouse 418, and a display 420. The display adaptor 422 interfaces with the communication  
22 bus 404 to receive display data from the processor 402 and converts the display data into

1 display commands for the display 420. The wireless interface 450 is configured to provide a  
2 communication interface between the destination computer platform 400 and a source  
3 computing platform.

4 Fig. 5 illustrates an exemplary block diagram of a software architecture executed by  
5 the destination computer system 400, as shown in Fig. 4, in accordance with the principles of  
6 the present invention. As shown in Fig. 5, the software architecture 500 includes at least an  
7 operating system 510, an API 515, a messaging program 520, and a synch module 530. The  
8 synch module 530 may include all the functionality of the synch module, 130 or 130', as  
9 discussed herein above. It should be readily apparent to those of ordinary skill in the art that  
10 Fig. 5 represents a generalized view of the software architecture and that other software  
11 components may be added or existing software components may be subtracted without  
12 departing from the spirit or scope of the present invention.

13 The operating system 510 of the destination computer system 400 may be configured  
14 to manage software programs or applications that provide functionality to the destination  
15 computer system 400. The operating system 510 may be further configured to provide an  
16 interface through the API 515 to the software applications and to the underlying hardware  
17 components, shown in Fig. 2, of the wireless device 200.

18 The messaging program 520 may be configured to provide the destination computer  
19 system 400 with the capability of receiving and transmitting instant messages between the  
20 user and other similarly-enabled users. The messaging program 520 may be configured to  
21 maintain a message history of conversations between the user and other similarly-enabled  
22 users. The message history may be stored on the main memory 406 of the destination

1 computer system 400, shown in Fig. 4. The messaging program 520 may be further  
2 configured to save the message history as a text file, a compressed data file or in a format  
3 configured to conserve space in the main memory 406. Examples of a messaging program  
4 may include AETHER INSTANT MESSAGING, JABBER, etc.

5 The synch module 530 of the destination computer system 400 may include the  
6 functionality of the synch module, 130 or 130', as discussed herein above. Moreover, the  
7 synch module 530 may be further configured to convert a transferred archived message  
8 history from a wireless device in response to a connection query. The connection query may  
9 be initiated by a synchronization module of a wireless device. The connection query may be  
10 configured to determine whether a communication channel is available between a wireless  
11 device and the destination computer system 400.

12 If the communication channel is available, the synch module 530 may be configured  
13 to receive an archived message history from a wireless device. The synch module 530 may be  
14 further configured to convert the format of the archived message history into a user-specified  
15 format. The synch module 530 may be further configured to provide a capability of selecting  
16 a data format (e.g., MICROSOFT WORD, HTML, XML, etc.,) for the converted archived  
17 message history. Moreover, the synch module 530 may be further configured to store the  
18 converted data, e.g., an archived message history, in a location specified by the user, e.g., the  
19 hard disk drive 410 of the destination computer system 400. Alternatively, the synch module  
20 530 of the destination computing platform may be further configured to execute similar  
21 functionality as the synch module 330 of the source computing platform, as described herein  
22 above, when the synch module 530 is a source of data.

1 Fig. 6 illustrates an exemplary flow diagram 600 of the synch module 330 shown in  
2 Fig. 3 functioning as a source synchronization module in accordance with the principles of the  
3 present invention. The synch module 330 may be configured to be in an idle state or inactive  
4 state, in step 605. The synch module 330 may receive a synchronization request from a user,  
5 in step 610. For example, the synchronization request may have been initiated by a menu  
6 option displayed on the display 220 or a combination of keypresses on the keypad 225 for the  
7 wireless device 200.

8 In step 615, the synch module 330 of source destination computing platform, e.g.,  
9 wireless device 200, may be configured to attempt to link or establish a connection or  
10 communication channel with a destination computer platform such as computer system 400,  
11 as shown in Fig. 4. The synch module 330 may link with the destination computer platform  
12 utilizing wired (e.g., RS-232, universal serial bus, etc.,) or wireless (e.g., infrared, radio  
13 frequency, Bluetooth, etc.,) protocols.

14 In step 620, the synch module 330 may be further configured to determine whether a  
15 connection has been established between the wireless device 200 and the destination  
16 computer platform. If the connection has not been established, the synch module 330 may be  
17 further configured to report that the destination computer system is unavailable to the user, in  
18 step 625. For example, the synch module 330 may display a text message on the display 220  
19 informing the user that the destination computer system is unavailable. The synch module  
20 330, in step 625, may be further configured to query the user whether the user would like to  
21 re-attempt to link with the destination computer via displaying a text message on the display  
22 220. If the user decides to re-attempt to connect to the destination computer, for example, by

1 pressing a combination of keypresses on the keypad 225, the synch module 330 may be  
2 further configured to return to step 615. Otherwise, the synch module 330 may be further  
3 configured to return to an idle state 605.

4 Returning to step 620, if the synch module 330 of the source computing platform is  
5 successful in connecting to the destination computer, the synch module 330 may be  
6 configured to send an activate message or signal to a destination synchronization module of  
7 the destination computer system, in step 635. The activate message may be configured to  
8 prepare the destination synchronization module for the receipt of data from the wireless  
9 device 200. The destination synchronization module may be configured to send an activated  
10 message response configured to inform the synch module 330 that the destination  
11 synchronization module has been activated.

12 In step 640, the synch module 330 of the wireless device 200 may be configured to  
13 transfer the archived message history to the destination synchronization module in response to  
14 receiving the activated message. The synch module 330 may be further configured to  
15 determine whether the data transfer is completed, in step 645. If the archived message  
16 history has been transferred, the synch module 330 may be further configured to return to the  
17 idle state 605. Otherwise, the synch module 330 may be further configured to enter a wait  
18 state, in step 650, and return to step 645 after the wait state has expired to determine if the  
19 data transfer has been completed. The length of the wait state may be a variable which may  
20 be set by the user, programmer, or manufacturer.

21 Fig. 7 illustrates an exemplary flow diagram 700 of the synch module 530 functioning  
22 as a destination synch module in accordance with the principles of the present invention. As

1 shown in Fig. 7, the synch module 530 of the destination computer system 120 may be  
2 configured to be in an idle state, in step 705. The destination computer system 120 may  
3 receive messages or signals from remote devices while in the idle state.

4 In step 710, if one of the messages is an activation message from a source computing  
5 platform, e.g., a wireless device, the synch module 530 may be further configured to establish  
6 a connection in step 715 with the requesting device. Otherwise, the synch module 530 may  
7 be further configured to return to the idle state of step 705.

8 Once the connection has been established, the synch module 550 may be further  
9 configured to transmit an activation response signal or message, in step 720. The activation  
10 signal may be in the form of a protocol handshake, a message packet, and the like.

11 In step 725, the synch module 530 may be further configured to receive data, e.g.,  
12 archived message history, from the source computing platform. The data is converted into a  
13 user-specified format in step 730. Alternatively, the synch module 530 may be configured to  
14 convert the received data after the source computing platform completes the transfer of the  
15 received data.

16 In step 735, the synch module 530 may be further configured store the converted data  
17 in a pre-determined location. The synch module 530 may be configured to previously provide  
18 a user interface, e.g., a menu option, a dialog box, a command line prompt, etc., whereby the  
19 user may specify a location to store the converted data.

20 In step 740, the synch module 530 may be further configured to transmit a  
21 synchronization completed signal or message to the source computing platform and the  
22 connection between the destination computer system 120 and a source device is severed or

1 ended, in step 745. The synch module 530 may be further configured to return to the idle  
2 state of step 705.

3 Fig. 8 illustrates an exemplary block diagram of a system architecture 800 of another  
4 embodiment of the present invention. As shown in Fig. 8, the system architecture 800  
5 includes a source computing platform 810, a message server 820 as an intermediate  
6 computing platform, a destination computing platform 830, a client computing platform 840,  
7 and a storage system 850. Although, for illustrative purposes only, Fig. 8 shows one source  
8 computing platform 810, one message server 820, one destination computing platform 830,  
9 and one client computing platform 840, it should be readily apparent to one of ordinary skill  
10 in the art that any number of devices may be used without departing from scope or spirit of  
11 the present invention.

12 The source computing platform 810 similar to the source computing platform device  
13 110 shown in Fig. 1 described hereinabove and thus only those features which are reasonably  
14 necessary for a complete understanding of this embodiment is described herein below. The  
15 message server 820 may be configured to provide a messaging service, e.g., AETHER  
16 INSTANT MESSAGING, JABBER, etc., to users. The message server 820 may be  
17 implemented using computing platforms such as SUN ENTERPRISE 10000 SERVER,  
18 SILICON GRAPHICS INCORPORATED 2100, HEWLETT PACKARD 9000, etc. The  
19 message server 820 may be configured to interface with computing platforms such as wireless  
20 devices through a wireless networks such as personal communication services network, a  
21 pager network, a cellular telephone network, etc., or a combination thereof.

22 The message server 820 may be further configured to provide an archiving or storage



1 service to source computing platforms, e.g., wireless devices such as text pagers, WAP-  
2 enabled telephones, etc., that are not equipped with sufficient memory to store data, e.g., an  
3 archived message history. For example, a text pager as the source computing platform 810  
4 may not contain enough memory to maintain a message history for a messaging session  
5 between users. As a result, the message server 820 may be configured to provide memory  
6 space to temporarily store data, e.g., the message history.

7 The message server 820 may be further configured to interface to the destination  
8 computing platform 830 via wired (e.g., local area networks, wide area networks, etc.,) means  
9 or wireless networks. The destination computing platform 830 is substantially similar to the  
10 destination computing platform 120 as shown in Fig. 1, and thus only those features which  
11 are reasonably necessary for a complete understanding of this embodiment is described herein  
12 below. The destination computing platform 830 may be configured to provide an archiving or  
13 storage service for users of the source computing platform 810. The destination computing  
14 platform 830 may be configured to convert received data, e.g., a message history, into a user-  
15 specified format and store the converted data in a pre-determined user location in the storage  
16 device 850 interfaced with the destination computing platform 830.

17 Fig. 9 illustrates an exemplary block diagram of a software architecture 900 of an  
18 embodiment of the present invention as implemented in the system architecture of Fig. 8. As  
19 shown in Fig. 9, the source computing platform 810, message server 820 and the destination  
20 computing platform 830 each contain a respective synch module, 910, 920, 930. Each of the  
21 synch modules 910, 920, 930 may be configured to include similar functionality as the synch  
22 module, 130 or 130'. Accordingly, each synch module may have the functionality of a source

1 and/or destination synch module depending on which computing platform is the  
2 source/destination of data as discussed hereinabove with respect to Figs. 1-7. Moreover, each  
3 synch module, 910, 920, 930, may include an intermediate synch module functionality. With  
4 regard to the intermediate synch module aspect, the synch module, 910, 920, 930, may be  
5 configured to provide interim processing of data from a source computing platform to a  
6 destination computing platform.

7 In particular, an intermediate computing platform may contain data related to a  
8 synchronization request. The sync module of the intermediate computing platform may be  
9 configured to combine any data related to the synchronization request into a single file on an  
10 intermediate computing platform.

11 The synch module of the intermediate computing platform may be further configured  
12 to determine if the intermediate computing platform is the final destination of the data from  
13 the source computing platform. If the intermediate computing platform is the final  
14 destination, the synch module may be configured to convert the transferred data into a data  
15 format specified by the user. The synch module may be further configured to send a  
16 completion signal or message to inform a source computing platform that the synchronization  
17 request has been completed. Otherwise, the synch module may be further configured to  
18 determine whether the destination computing platform is available.

19 If the destination computing platform is unavailable, the synch module of the  
20 intermediate computing platform may temporarily store the transferred data while periodically  
21 attempting to establish a communication channel with the destination computing platform.  
22 Otherwise, the synch module may be further configured to establish a communication channel

1 with the destination computing platform. The synch module may be further configured to  
2 activate a synch module of the destination computing platform in response to the  
3 establishment of a communication channel between the intermediate and destination  
4 computing platforms. The synch module of the intermediate computing platform may be  
5 further configured to transfer the designated data to the destination computing platform.

6 As an example of an embodiment of the present invention, if the source computing  
7 platform 810 is configured as a source of data, the synch module 910, functioning as a source  
8 synchronization module, may be configured to transfer the saved or archived message history  
9 from a messaging program 916 in the source computing platform 810 to a destination  
10 computer system in response to a synchronization request. The synchronization request may  
11 be initiated as a menu option, an icon or manually initiated through a keypad on the source  
12 computing platform 810.

13 The synch module 910 of the source computing platform 810 may be further  
14 configured to determine whether a communication channel is available over an interface  
15 (wired/wireless) to an intermediate/destination computer system through an API 912 of an  
16 operating system 914. In the event that the synch module 910 is unable to open a  
17 communication channel to the intermediate/destination computer system, the synch module  
18 910 may be further configured to indicate that the synchronization could not be performed.

19 If the communication channel is available, the synch module 920 of the intermediate  
20 computing platform, such as the message server 820, may be interfaced through an API 922  
21 of an operating system 924. The API 922 may be further configured to interface with a  
22 messaging program 926. The messaging program 926 may be configured to provide instant

1 messaging services for users of the system architecture 800. The messaging program 926  
2 may be implemented by AETHER INTELLIGENT MESSAGING, JABBER, etc.

3 The synch module 920 of the message server 820 may be further configured to receive  
4 data, e.g., a message history, from the source computing platform 810. The synch module  
5 920 may be further configured to combine any data related to the received data and to  
6 determine whether the message server 820 is the destination for the data. If the message  
7 server 820 is final destination for the data, the synch module 920 may be further configured to  
8 store the data in predetermined location.

9 Otherwise, the synch module 920 may be further configured to establish a connection  
10 with the destination computing platform 830 through an API 932 of an operating system 934.

11 The API 932 may be further configured to interface with a messaging program 936. If the  
12 connection is not enabled, the synch module 920 may temporarily store the combined data in  
13 memory location and attempt to establish a connection with the destination computing  
14 platform 830 at a later time. Otherwise, once the connection is established, the synch module  
15 920 of the message server 820 may be configured to transfer the combined data to the  
16 destination computing platform 830.

17 The synch module 930 of the destination computing platform 830 may be configured  
18 to convert received data, e.g., a message history, to a user-specified format. The synch  
19 module 930 may be further configured to store the converted data in a location specified by  
20 the user in the storage device 850. Alternatively, the converted data may be stored in a  
21 memory of a computer system 840 by the destination synch module 930.

22 Fig. 10 illustrates an exemplary flow diagram 1000 for a source synchronization

1 module in the system architecture as shown in Figs. 8-9. As shown in Fig. 10, a source  
2 synchronization module may be configured to be in an idle state, in step 1005. In step 1010, a  
3 synchronization may be initiated by receipt of a synchronization request. The  
4 synchronization request may be initiated by a menu option, a command line prompt, a  
5 combination of keypresses, etc.

6 Once the synchronization is initiated, the source synchronization module may be  
7 further configured to establish a connection in step 1015. The source synchronization module  
8 may initiate a connection protocol or transmit a message to connect with a destination  
9 synchronization module. If the connection is not enabled, the source synchronization module  
10 may be further configured to store the data, e.g., a message history, on the source platform  
11 such as a wireless device, a computing platform, etc.

12 Otherwise, when the connection is established in step 1020, the source  
13 synchronization module may be further configured to activate a destination synchronization  
14 module, in step 1025. The destination synchronization module may be activated as part of a  
15 protocol handshake, receiving an activation signal or message, etc.

16 In step 1030, the source synchronization module may be further configured to transfer  
17 data, e.g., a message history, in response to the activation of the destination synchronization  
18 module. In step 1035, the source synchronization module may be further configured to  
19 determine if the data transfer is completed. If the data transfer is not completed, the source  
20 synchronization module may be further configured to enter a wait state, in step 1040.

21 Returning to step 1035, if the data transfer is complete, the source synchronization  
22 module may be configured to return to the idle state of step 1005.

1           Returning to step 1020, if the connection cannot be established, the source  
2           synchronization module may be further configured to temporarily store the data, e.g., a chat  
3           history, on the source computing platform, in step 1045. The source synchronization module,  
4           in step 1050 may be further configured to report the unavailability of the destination  
5           computing platform.

6           In step 1055, the source synchronization module may be further configured to prompt  
7           a user for a re-attempt to establish communication with the destination computing platform.  
8           If the there is a re-attempt, the source synchronization module may be configured to return to  
9           step 1015. Otherwise, the source synchronization module may be configured to return to the  
10          idle state of step 1005.

11          Figs. 11A-C, together, illustrate an exemplary flow diagram 1100 for a  
12          destination/intermediate synchronization module in the system architecture as shown in Figs.  
13          8-9. As shown in Fig. 11A, the destination/intermediate synchronization module ("DISM")  
14          may be configured to be in an idle state in step 1105. The DISM may be further configured to  
15          determine whether a connection request has been received from a source synchronization  
16          module, in step 1110. This determination may be implemented asynchronously or on a  
17          periodic basis. If a connection request has not been received, the DISM may be configured to  
18          return to the idle state of 1105.

19          Otherwise, if a connection request has been received in step 1110, the DISM may be  
20          further configured to attempt to establish a link or connection with the source synchronization  
21          module, in step 1115. The connection may be implemented using handshake signals, TCP/IP,  
22          or similar protocols.

1 In step 1120, the DISM may be configured to receive an activation request from a  
2 source synchronization module. The activation request may be part of a handshake protocol,  
3 another received message or similar notification. Once activated, the DISM may be further  
4 configured to receive data, e.g., a message history, from the source synchronization module in  
5 step 1125.

6 In step 1130, the DISM may be configured to combine the received data, e.g., a  
7 message history, with any existing message history cached or temporarily stored on the  
8 current computing platform. A source computing platform may not contain enough memory  
9 space to store a complete message history. Accordingly, the current computing platform may  
10 be the message server 820 which may be configured to maintain a partial or full message  
11 history in a memory.

12 In step 1135, the DISM may be configured to determine if the current platform is the  
13 destination platform for the data from the source synchronization module. The data received  
14 may contain a destination code, e.g., a destination address, which may designate the  
15 destination computing platform. Each synchronization module may be assigned a location  
16 code where the DISM may compare the destination code with the location of the destination  
17 computing platform.

18 If the comparison is true, in step 1140 with reference to Fig. 11B, the DISM may be  
19 configured to convert the received data, e.g., a message history, into a predetermined or pre-  
20 selected format. The format may be configured by a user of the destination/intermediate  
21 computer system via a menu option, a command line prompt, an icon, etc. The DISM may  
22 convert the data in response to receipt of a signal indicating that all of the data from the

1 source computing platform or the DISM may convert the data as the data is received from the  
2 source computing platform.

3 In step 1145, the DISM may be configured to store the converted data in a  
4 predetermined location. The location may be specified by a user of the  
5 destination/intermediate computer system via a menu option, a command line prompt, an  
6 icon, etc. The DISM may be further configured to send a completion signal or message to the  
7 source synchronization module in response to storing the converted data, in step 1150.  
8 Subsequently, the DISM may be configured to return to the idle state of 1105 with reference  
9 to Fig. 11A.

10 Returning to step 1135, if the current computing platform is not the destination  
11 platform, the DISM may be configured to determine if the destination computing platform is  
12 available, in step 1155. If the destination computing platform is not available, the DISM may  
13 be configured to temporarily store the converted data in an available memory location, in step  
14 1160. The DISM may be further configured to transmit a signal or message to the source  
15 computing platform that the destination computing platform is currently unavailable, in step  
16 1165.

17 In step 1170, the DISM may be further configured to prompt a user for a re-attempt to  
18 establish communication with the destination computing platform. If there is a re-attempt,  
19 the DISM may asynchronously or periodically attempt to establish a connection with the  
20 destination computing platform by returning to step 1155. Otherwise, the DISM may be  
21 configured to return to the idle state of step 1105.

22 Returning to step 1155, if the destination computing platform is available and the



1 connection is established, the DISM may be further configured to activate the destination  
2 synch module of the destination computing platform, in step 1175 with reference to Fig. 11C.

3 The destination synchronization module may be activated as part of a protocol handshake,  
4 receiving an activation signal or message, etc. The DISM may be further configured to  
5 transfer the data to the destination computing platform in step 1180.

6 The DISM may be further configured to wait for a completion signal or message from  
7 the completion of the data transfer in step 1185. If the message is not received, the source  
8 synchronization module returns to the wait state of step 1190. Otherwise, the source  
9 synchronization module may be further configured to return to the idle state of 1105 of Fig.  
10 11A.

11 The present invention may be performed as a computer program. The computer  
12 program may exist in a variety of forms both active and inactive. For example, the computer  
13 program can exist as software program(s) comprised of program instructions in source code,  
14 object code, executable code or other formats; firmware program(s); or hardware description  
15 language (HDL) files. Any of the above can be embodied on a computer readable medium,  
16 which include storage devices and signals, in compressed or uncompressed form. Exemplary  
17 computer readable storage devices include conventional computer system RAM (random  
18 access memory), ROM (read-only memory), EPROM (erasable, programmable ROM),  
19 EEPROM (electrically erasable, programmable ROM), and magnetic or optical disks or tapes.  
20 Exemplary computer readable signals, whether modulated using a carrier or not, are signals  
21 that a computer system hosting or running the present invention can be configured to access,  
22 including signals downloaded through the Internet or other networks. Concrete examples of

1 the foregoing include distribution of executable software program(s) of the computer program  
2 on a CD ROM or via Internet download. In a sense, the Internet itself, as an abstract entity, is  
3 a computer readable medium. The same is true of computer networks in general.

4 What has been described and illustrated herein are preferred embodiments of the  
5 invention along with some of their variations. The terms, descriptions and figures used herein  
6 are set forth by way of illustration only and are not meant as limitations. Those skilled in the  
7 art will recognize that many variations are possible within the spirit and scope of the  
8 invention, which is intended to be defined by the following claims -- and their equivalents --  
9 in which all terms are meant in their broadest reasonable sense unless otherwise indicated.